

REMARKS

After entry of the above amendments, claims 1-61 will be pending in the present application. Claims 62-63 have been cancelled. Claims 1, 31, and 61 have been amended to incorporate elements from cancelled claims, to explicitly recite what was implicit, to further clarify the invention, and/or to correct informalities. Support for the claim amendments can be found in the specification, figures, and the claims as originally filed. Applicant reserves the right to pursue any cancelled claim and any amended claim in its original form in a continuation application. No new matter has been added.

Objections to the Specification

The Examiner objected to the Specification for omitting serial numbers and filing dates of applications incorporated by reference. Applicant has amended the Specification to include such information. Hence, withdrawal of the objections to the Specification is respectfully requested.

Claim Rejections

Claims 1-2 and 4-10 have been rejected under 35 U.S.C. § 102(b) as being anticipated by U.S. Patent No. 5,758,333 to Bauer et al. (hereinafter "Bauer"). Claim 3 has been rejected under 35 U.S.C. § 103(a) as being unpatentable over Bauer, in view of U.S. Patent No. 6,058,389 to Chandra et al. (hereinafter "Chandra").

Claim 1, as amended, recites:

1. A method for preserving data constraints during parallel application of asynchronous replication transactions in a database system, the method comprising:
 - (a) receiving and examining a transaction message comprising one or more row changes;
 - (b) determining whether at least one row change in the transaction message is affected by a data constraint;
 - (c) responsive to at least one row change in the transaction message being affected by a data constraint, determining whether there is a constraint violation between the at least one row change in the transaction message and a row change in at least one preceding non-completed transaction message;
 - (d) responsive to there being a constraint violation between the at least one row change in the transaction message and the row change in the at least one preceding non-completed transaction message, holding the transaction message until application of the at least one preceding non-completed transaction message is completed;
 - (e) responsive to application of the at least one preceding non-completed transaction message being completed, placing the transaction message on a work queue; and
 - (f) subsequent to placing the transaction message on the work queue, applying the transaction message in parallel with one or more other transaction messages on the work queue.

In the Office action, the Examiner states:

Bauer discloses (a) receiving and examining a transaction message; (b) determining if at least one row change in the transaction message has data constraints [col 4, lines 10-20, col 4, lines 40-50] (c) if so, determining if the at least one row change in the transaction message has a constraint violation with a row change in at least one preceding non-completed transaction message [col 5, lines 38-44] and (d) if so, holding the transaction message until the at least one preceding non-completed transaction message completes [rejected transactions are returned to the user, col 5, line 60 through col 6, line 10].

(September 15, 2006 Office action, pg. 3).

The last passage of Bauer cited by the Examiner states:

In the preparation phase, the calls of the access procedures are recorded and any amendments of the data affected by the access are prepared. In the

preparation phase, the user has the possibility of rejecting the transaction by calling up the procedure ABORT.sub.-- TA.

Calling up the procedure DO.sub.-- TA at the same time has the effect of starting the activation phase, which is controlled exclusively by the access control system and therefore runs independently of user calls. In the activation phase, consequently no further calls for individual accesses can be taken into consideration.

In the preparation phase, the coordination of a transaction is carried out, i.e. all the collision cases with other parallel transactions are identified and coordinated. The coordination in this case takes place by those transactions which were started earlier being allowed to continue, whereas other parallel transactions are rejected by a corresponding negative acknowledgement being returned to the user. Since all collision cases are consequently resolved in the preparation phase, transactions can be carried out independently of one another in the activation phase. FIG. 4 shows the structure of the central access control system CTCC. The central access control system CTCC comprises a transaction management system TMS, a generations management system GMS, a resources management system RMS and a collision management system CMS.

(Col. 5, ln. 50 to col. 6, ln. 10 of Bauer) (emphasis added).

As pointed out by the Examiner, the cited passage of Bauer only discusses rejecting certain parallel transactions. Claim 1, however, recites "responsive to there being a constraint violation between the at least one row change in the transaction message and the row change in the at least one preceding non-completed transaction message, holding the transaction message until application of the at least one preceding non-completed transaction message is completed" (emphasis added). It does not recite that the transaction message is rejected.

Bauer also states:

The access control system controls access by users to the data management system EDB and thereby ensure that an updating operation takes the data management system from a consistent initial state into a consistent end state. To ensure this consistency, the access control system handles access sequences (transactions or reading sequences) in an integral (atomic) way, i.e. a transaction is, for example, either carried out in its entirety or rejected in its entirety.

(Col. 4, lns. 8-15 of Bauer) (emphasis added).

Hence, in Bauer, once a transaction is rejected, it is not carried out. Claim 1, in contrast, further recites “responsive to application of the at least one preceding non-completed transaction message being completed, placing the transaction message on a work queue” and “subsequent to placing the transaction message on the work queue, applying the transaction message in parallel with one or more other transaction messages on the work queue” (emphasis added). Therefore, rejection of the transaction in Bauer cannot be construed as disclosing “holding the transaction message until application of the at least one preceding non-completed transaction message is completed,” as recited in claim 1.

Chandra does not cure the deficiencies of Bauer. Therefore, even if Bauer and Chandra were combined, the combination would neither teach nor suggest all of the elements recited in claim 1. Accordingly, based at least on the reasons above, Applicant respectfully submits that claim 1, and the claims that depend therefrom, are not anticipated by Bauer and are patentable over Bauer in view of Chandra. Since claims 31 and 61 each recite elements similar to those of claim 1, it is respectfully submitted that those claims, and the claims that depend therefrom, are not anticipated by Bauer and are patentable over Bauer in view of Chandra for at least the same reasons.

In the Office action, the Examiner also states:

Regarding claims 11-63, examiner maintains that such claims can be rejected on a similar basis to claims 1-10 above.

(September 15, 2006 Office action, pg. 5).

Claims 18, however, recites:

18. A method for preserving data constraints during parallel apply in asynchronous transaction replication in a database system, comprising:

- identifying a transaction message as a cascade delete;
- determining that a source of the transaction message is not a leaf table;
- placing each subscription for the transaction message onto a stack and placing row operations for each subscription into a reorder list, wherein the subscriptions are placed onto the stack in order of execution, wherein the row operations are placed into the reorder list in the order of execution; and
- adding the row operations for each subscription in the stack back to the transaction message, wherein the row operations are added in a reverse order of execution, wherein the subscriptions are added in the reverse order of execution.

In addition, claims 21 recites:

21. A method for preserving data constraints during parallel apply in asynchronous transaction replication in a database system, comprising:

- receiving a message to perform an initial load of a target table;
- determining that the target table is a child table of referential integrity constraints;
- saving the referential integrity constraints for the target table;
- dropping the referential integrity constraints from the target table;
- loading the target table in parallel with a loading of a parent table of the referential integrity constraints;
- begin applying change data to the target table once loading is done;
- waiting for the parent table to finish loading, if the parent table has not yet finished loading; and
- adding the referential integrity constraints back into the target table.

Further, claim 26 recites:

26. A method for preserving data constraints during parallel apply in asynchronous transaction replication in a database system, comprising:

- receiving a message to perform an initial load of a target table;
- determining that the target table is a parent table of referential integrity constraints;
- saving the referential integrity constraints for a child table of the target table;

- (d) dropping the referential integrity constraints from the child table;
- (e) loading the target table in parallel with a loading of the child table;
- (f) begin applying change data to the target table once loading is done;
- (g) waiting for the child table to finish loading, if the child table has not yet finished loading; and
- (h) adding the referential integrity constraints back into the child table.

The elements of claims 18, 21, and 26 are not recited in claim 1 or the claims that depend therefrom. Since the Examiner does not cite any passage of Bauer or Chandra as disclosing the elements of claims 18, 21, and 26, Applicant respectfully submits that claims 18, 21, and 26, and the claims that depend therefrom, are patentable over Bauer in view of Chandra. Since claims 48, 51, and 56 recite elements similar to those of claims 18, 21, and 26, respectively, it is respectfully submitted that claims 48, 51, and 56, and the claims that depend therefrom, are patentable over Bauer in view of Chandra for at least the same reasons.

CONCLUSION

On the basis of the above remarks, reconsideration and allowance of the claims is believed to be warranted and such action is respectfully requested. If the Examiner has any questions or comments, the Examiner is respectfully requested to contact the undersigned at the number listed below.

Respectfully submitted,
SAWYER LAW GROUP LLP



Erin C. Ming
Attorney for Applicant(s)
Reg. No. 47,797
(650) 475-1449

Dated: December 14, 2006